

日本国特許庁  
JAPAN PATENT OFFICE

08.11.2004

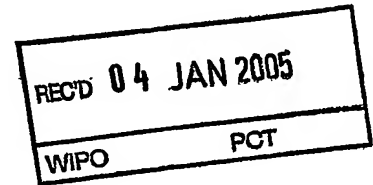
別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日  
Date of Application: 2003年11月18日

出願番号  
Application Number: 特願2003-387351  
[ST. 10/C]: [JP 2003-387351]

出願人  
Applicant(s): 松下電器産業株式会社



BEST AVAILABLE COPY

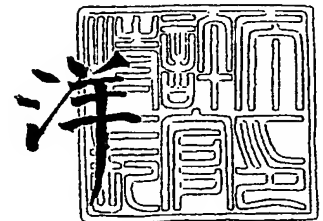
CERTIFIED COPY OF  
PRIORITY DOCUMENT

PRIORITY DOCUMENT  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH  
RULE 17.1(a) OR (b)

2004年12月16日

特許庁長官  
Commissioner,  
Japan Patent Office

小川



【書類名】 特許願  
【整理番号】 2037950005  
【あて先】 特許庁長官殿  
【国際特許分類】 G06F 12/08 310  
【発明者】  
    【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内  
    【氏名】 岡林 はづき  
【発明者】  
    【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内  
    【氏名】 中西 龍太  
【発明者】  
    【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内  
    【氏名】 田中 哲也  
【発明者】  
    【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内  
    【氏名】 宮阪 修二  
【特許出願人】  
    【識別番号】 000005821  
    【氏名又は名称】 松下電器産業株式会社  
【代理人】  
    【識別番号】 100109210  
    【弁理士】  
    【氏名又は名称】 新居 広守  
【手数料の表示】  
    【予納台帳番号】 049515  
    【納付金額】 21,000円  
【提出物件の目録】  
    【物件名】 特許請求の範囲 1  
    【物件名】 明細書 1  
    【物件名】 図面 1  
    【物件名】 要約書 1  
    【包括委任状番号】 0213583

**【書類名】 特許請求の範囲****【請求項 1】**

キャッシュの単位データを保持するキャッシュエントリー毎にアクセス順序を示す順序データを保持し、最も古い順序を示すキャッシュエントリーをリプレースするキャッシュメモリであって、

前記順序データを実際のアクセス順序に反して改変する改変手段と、

改変後の順序データに基づいてリプレースすべきキャッシュエントリーを選択する選択手段と、

を備えることを特徴とするキャッシュメモリ。

**【請求項 2】**

前記改変手段は、

プロセッサから指定されたアドレス範囲に属するデータを保持するキャッシュエントリーを特定する特定手段と、

特定されたキャッシュエントリーの順序データを実際のアクセス順序に反して最古化する最古化手段と

を備えることを特徴とする請求項 1 記載のキャッシュメモリ。

**【請求項 3】**

前記特定手段は、

前記アドレス範囲の先頭アドレスがラインデータの途中を指す場合、当該先頭アドレスを、前記アドレス範囲に含まれる先頭のラインを指すスタートラインアドレスに変換する第 1 変換手段と、

前記アドレス範囲の末尾アドレスがラインデータの途中を指す場合、当該末尾アドレスを、前記アドレス範囲に含まれる末尾のラインを指すエンドラインアドレスに変換する第 2 変換手段と、

前記スタートラインアドレスからエンドラインアドレスまでの各ラインアドレスに対応するデータを保持するキャッシュエントリーがあるか否かを判定する判定手段と

を備えることを特徴とする請求項 2 記載のキャッシュメモリ。

**【請求項 4】**

前記最古化手段は、アクセス順序が最も古いことを示す最古化フラグを順序データに付加する

ことを特徴とする請求項 3 記載のキャッシュメモリ。

**【請求項 5】**

前記選択手段は、キャッシュミス時に、最古化フラグが付加されたキャッシュエントリーが存在する場合、そのキャッシュエントリーをリプレース対象として選択し、最古化フラグが付加されたキャッシュエントリーが存在しない場合、順序データに従ってリプレース対象とすべきキャッシュエントリーを選択する

ことを特徴とする請求項 4 記載のキャッシュメモリ。

**【請求項 6】**

前記キャッシュエントリーは、アクセス順序が古いか新しいかを示す 1 ビットの順序フラグを前記順序データとして有し、

前記選択手段は、最古化フラグが付加されたキャッシュエントリーが存在しない場合、順序フラグが古いことを示すキャッシュエントリーをリプレース対象として選択する

ことを特徴とする請求項 5 記載のキャッシュメモリ。

**【請求項 7】**

前記改変手段は、1つのキャッシュエントリーのアクセス順序をN番目にするよう前記順序データを改変し、

前記Nは、(a)アクセス順序が最も古い順であることを示す番号、(b)アクセス順序が最も新しいことを示す番号、(c)アクセス順序が最も古い順序からN番目であることを示す番号、(d)アクセス順序が最も新しい順序からN番目であることを示す番号の何れかである

ことを特徴とする請求項1記載のキャッシュメモリ。

【請求項8】

前記改変手段は、

アクセス順序の改変指示付きメモリアクセス命令が実行されたことを検出する命令検出手段と、

当該命令によってアクセスがなされたキャッシュエントリーに対して順序データを書き替える書き換え手段と

を備えることを特徴とする請求項1記載のキャッシュメモリ。

【請求項9】

前記改変手段は、

プロセッサから指定されたアドレス範囲を保持する保持手段と、

保持手段に保持されたアドレス範囲に対応するデータを保持するキャッシュエントリーを探索する探索手段と、

探索手段により探索されたキャッシュエントリーのアクセス順序をN番目とするよう順序データを書き換える書き換え手段と

を備えることを特徴とする請求項1記載のキャッシュメモリ。

【請求項10】

キャッシュの単位データを保持するキャッシュエントリー毎にアクセス順序を示す順序データを保持し、最も古い順序を示すキャッシュエントリーをリプレースするキャッシュメモリの制御方法であって、

前記順序データを実際のアクセス順序に反して改変する改変ステップと、

改変後の順序データに基づいてリプレースすべきキャッシュエントリーを選択する選択ステップと

を有することを特徴とする制御方法。

## 【書類名】明細書

## 【発明の名称】 キャッシュメモリ及びその制御方法

## 【技術分野】

## 【0001】

本発明は、プロセッサのメモリアクセスを高速化するためのキャッシュメモリおよびその制御方法に関する。

## 【背景技術】

## 【0002】

近年のマイクロプロセッサでは、例えば、SRAM (Static Random Access Memory) 等から成る小容量で高速なキャッシュメモリをマイクロプロセッサの内部、もしくはその近傍に配置し、データの一部をキャッシュメモリに記憶することによって、マイクロプロセッサのメモリアクセスを高速化させている。

上記コンピュータシステムでは、中央処理装置からキャッシュメモリへのリードアクセスまたはライトアクセスがミスヒットした場合に、主記憶装置から新たに読み出したデータの一部が、エントリ（登録項目）としてキャッシュメモリの空きブロックに格納される。この時、空きブロックが存在しない場合には、複数のブロックの何れか1つを選択し、選択されたブロックに格納されているエントリを主記憶装置に戻してブロック内を空き状態にし、この空きブロックに新たに読み出したデータを格納するエントリ置換処理が必要になる。上記エントリ置換処理では、最も以前に参照したデータを格納しているブロックを選択する手法、すなわち、LRU (Least Recently Used) デコード方式が一般的に採用されている。このLRUデコード方式によってキャッシュメモリの使用効率が向上し、その結果、マイクロプロセッサの実行速度が向上する。

## 【0003】

マイクロプロセッサが処理するプログラムの中には、アクセス頻度は少ないものの、ひとたび起動された場合には高速に処理しなければならないような特殊な処理と、アクセス頻度は多いが、実行速度がそれほど要求されないような処理とが存在する。

## 【0004】

そこで、これに対応するために例えば特許文献1等の従来技術では、キャッシュメモリにフリーズ機能を設けている。フリーズ機能は、アクセスは少ないものの、ひとたび起動された場合には高速に処理しなければならないようなプログラムを予めキャッシュメモリ内にコピーしておき、その領域を書き換え禁止にしておく機能である。この機能を有することで、コンピュータシステムは、必要な時にプログラムをキャッシュメモリから読み出して実行することができ、これにより実行時間が短縮する。また、パージ機能は、アクセス頻度は多いが、実行速度がそれほど要求されないようなプログラムやデータをキャッシュメモリ内に保存しておくことなく、その領域を解放する機能である。この機能を有することで、キャッシュメモリに余裕ができ、優先度の高い他のプログラムやデータをキャッシュメモリに取り込むことができ、これにより、キャッシュメモリの利用効率が向上し、総合的な実行時間が短縮する。

【特許文献1】特開2000-200221号公報

## 【発明の開示】

## 【発明が解決しようとする課題】

## 【0005】

しかしながら、フリーズ機能もパージ機能も、アクセス頻度の高いデータをキャッシュメモリに留めたり、アクセス頻度の低いデータをキャッシュメモリから追い出すことを可能としているが、フリーズ機能及びパージ機能の制御のために複雑な回路を必要とするという問題がある。

本発明は、複雑な回路を設けることなく、アクセス頻度の低いデータを他のアクセス頻度の高いデータよりも優先的にリプレース対象とするキャッシュメモリを提供することを目的とする。

## 【課題を解決するための手段】

**【0006】**

上記課題を解決するため本発明のキャッシュメモリは、キャッシュの単位データを保持するキャッシュエントリー毎にアクセス順序を示す順序データを保持し、最も古い順序を示すキャッシュエントリーをリプレースするキャッシュメモリであって、前記順序データを実際のアクセス順序に反して改変する改変手段と、改変後の順序データに基づいてリプレースすべきキャッシュエントリーを選択する選択手段とを備える。

**【0007】**

この構成によれば、例えば、アクセス頻度の低いデータの順序データをアクセス順序が最も古いように改変することにより、アクセス頻度の低いデータを他のアクセス頻度の高いデータよりも優先的にリプレース対象とすることができ、アクセス頻度の高いデータの順序データをアクセス順序が最新又は最も古くないことを示すように改変することにより、アクセス頻度の高いデータがリプレースされることを防止することができる。

**【0008】**

ここで、前記改変手段は、プロセッサから指定されたアドレス範囲に属するデータを保持するキャッシュエントリーを特定する特定手段と、特定されたキャッシュエントリーの順序データを実際のアクセス順序に反して最古化する最古化手段とを備える構成としてもよい。

この構成によれば、プロセッサからこれ以上読み出しも書き込みもなされないキャッシュエントリーのアクセス順序を最古にすれば、当該キャッシュエントリーは真つ先にリプレース対象として選択される。これによりアクセス頻度の低いデータがキャッシュメモリに居残ることによるキャッシュミスの誘発を低減することができる。

**【0009】**

ここで、前記特定手段は、前記アドレス範囲の先頭アドレスがラインデータの途中を指す場合、当該先頭アドレスを、前記アドレス範囲に含まれる先頭のラインを指すスタートラインアドレスに変換する第1変換手段と、前記アドレス範囲の末尾アドレスがラインデータの途中を指す場合、当該末尾アドレスを、前記アドレス範囲に含まれる末尾のラインを指すエンドラインアドレスに変換する第2変換手段と、前記スタートラインアドレスからエンドラインアドレスまでの各ラインアドレスに対応するデータを保持するキャッシュエントリーがあるか否かを判定する判定手段とを備える構成としてもよい。

**【0010】**

この構成によれば、プロセッサは、前記アドレス範囲として、キャッシュメモリのラインサイズ及びライン境界のアドレスとは無関係に任意のアドレスから任意のアドレス（又は任意のサイズ）を指定することができる。つまり、プロセッサにおいてキャッシュメモリのラインサイズ及びライン境界のアドレスを管理する必要がないので、キャッシュメモリ管理のための負荷を解消することができる。

**【0011】**

ここで、前記最古化手段は、アクセス順序が最も古いことを示す最古化フラグを順序データに付加する構成としてもよい。

この構成によれば、LRU方式におけるアクセス順序を示す順序データを直接改変することなく、間接的にWフラグを付加することによってアクセス順序を改変しているので、複雑なハードウェア回路を追加することなく実現することができる。

**【0012】**

ここで、前記選択手段は、キャッシュミス時に、最古化フラグが付加されたキャッシュエントリーが存在する場合、そのキャッシュエントリーをリプレース対象として選択し、最古化フラグが付加されたキャッシュエントリーが存在しない場合、順序データに従ってリプレース対象とすべきキャッシュエントリーを選択する構成としてもよい。

ここで、前記キャッシュエントリーは、アクセス順序が古いか新しいかを示す1ビットの順序フラグを前記順序データとして有し、前記選択手段は、最古化フラグが付加されたキャッシュエントリーが存在しない場合、順序フラグが古いことを示すキャッシュエントリーをリプレース対象として選択する構成としてもよい。

**【0013】**

この構成によれば、順序データが1ビットの順序フラグでよいので、アクセス順序データのデータ量が少ないこと及び更新が簡単であることからハードウェア規模を小さくすることができる。

**【0014】**

ここで、前記改変手段は、1つのキャッシュエントリーのアクセス順序をN番目にするよう前記順序データを改変し、前記Nは、(a)アクセス順序が最も古い順であることを示す番号、(b)アクセス順序が最も新しいことを示す番号、(c)アクセス順序が最も古い順序からN番目であることを示す番号、(d)アクセス順序が最も新しい順序からN番目であることを示す番号の何れかであるように構成してもよい。

**【0015】**

ここで、前記改変手段は、アクセス順序の改変指示付きメモリアクセス命令が実行されたことを検出する命令検出手段と、当該命令によってアクセスがなされたキャッシュエントリーに対して順序データを書き替える書き換え手段とを備える構成としてもよい。

ここで、前記改変手段は、プロセッサから指定されたアドレス範囲を保持する保持手段と、保持手段に保持されたアドレス範囲に対応するデータを保持するキャッシュエントリーを探索する探索手段と、探索手段により探索されたキャッシュエントリーのアクセス順序をN番目とするよう順序データを書き換える書き換え手段とを備える構成としてもよい。

また、本発明のキャッシュメモリの制御方法についても上記と同様の手段、作用を有する。

**【発明の効果】****【0016】**

本発明のキャッシュメモリによれば、アクセス頻度の低いデータを他のアクセス頻度の高いデータよりも優先的にリプレース対象とすることができ、また、アクセス頻度の高いデータがリプレースされることを防止することができる。

例えば、プロセッサからこれ以上読み出しも書き込みもなされないキャッシュエントリーのアクセス順序を最古にすれば、当該キャッシュエントリーは真っ先にリプレース対象として選択される。これによりアクセス頻度の低いデータがキャッシュメモリに居残ることによるキャッシュミスの誘発を低減することができる。

**【0017】**

また、プロセッサにおいてキャッシュメモリのラインサイズ及びライン境界のアドレスを管理する必要がないので、プロセッサにおけるキャッシュメモリ管理のための負荷を解消することができる。

**【発明を実施するための最良の形態】****【0018】****(実施の形態1)****<全体構成>**

図1は、本発明の実施の形態1におけるプロセッサ1、キャッシュメモリ3、メモリ2を含むシステムの概略構成を示すブロック図である。同図のように、本発明のキャッシュメモリ3は、プロセッサ1およびメモリ2を有するシステムに備えられる。

**【0019】**

キャッシュメモリ3は、いわゆるLRU方式によってアクセス順序の古いキャッシュエントリーをリプレースするリプレース制御を前提としている。本実施の形態のキャッシュメモリ3は、リプレース対象を決定するためのアクセス順序を示す順序データを、アクセス順序に反して改変することにより、アクセス頻度の低いデータを保持するキャッシュエントリーをリプレース対象としてキャッシュメモリから追い出すよう構成されている。具体的には、アクセス順序が最古であるとみなすことを示すウィークフラグWをキャッシュエントリーに付加することによって、順序データを間接的に改変している。これにより順序データを直接改変する複雑な回路を不要としている。

## 【0020】

## &lt;キャッシュメモリの構成&gt;

以下、キャッシュメモリ3の具体例として、4ウェイ・セット・アソシエティブ方式のキャッシュメモリに本発明を適用した場合の構成について説明する。

## 【0021】

図2は、キャッシュメモリ3の構成例を示すブロック図である。同図のように、キャッシュメモリ3は、アドレスレジスタ20、メモリI/F21、デコーダ30、4つのウェイ31a~31d（以下ウェイ0~3と略す）、4つの比較器32a~32d、4つのアンド回路33a~33d、オア回路34、セクタ35、36、デマルチプレクサ37、制御部38を備える。

## 【0022】

アドレスレジスタ20は、メモリ2へのアクセスアドレスを保持するレジスタである。このアクセスアドレスは32ビットであるものとする。同図に示すように、アクセスアドレスは、最上位ビットから順に、21ビットのタグアドレス、4ビットのセットインデックス（図中のSI）、5ビットのワードインデックス（図中のWI）を含む。ここで、タグアドレスはウェイにマッピングされるメモリ中の領域（そのサイズはセット数×ブロックである）を指す。この領域のサイズは、タグアドレスよりも下位のアドレスビット（A10~A0）で定まるサイズつまり2kバイトであり、1つのウェイのサイズでもある。セットインデックス（SI）はウェイ0~3に跨る複数セットの1つを指す。このセット数は、セットインデックスが4ビットなので16セットある。タグアドレスおよびセットインデックスで特定されるキャッシュエントリは、リプレース単位であり、キャッシュメモリに格納されている場合はラインデータ又はラインと呼ばれる。ラインデータのサイズは、セットインデックスよりも下位のアドレスビットで定まるサイズつまり128バイトである。1ワードを4バイトとすると、1ラインデータは32ワードである。ワードインデックス（WI）は、ラインデータを構成する複数ワード中の1ワードを指す。アドレスレジスタ20中の最下位2ビット（A1、A0）は、ワードアクセス時には無視される。

## 【0023】

メモリI/F21は、キャッシュメモリ3からメモリ2へのデータのライトバックや、メモリ2からキャッシュメモリ3へのデータのロード等、キャッシュメモリ3からメモリ2をアクセスするためのI/Fである。

デコーダ30は、セットインデックスの4ビットをデコードし、4つのウェイ0~3に跨る16セット中の1つを選択する。

## 【0024】

4つのウェイ0~3は、同じ構成を有数する4つのウェイであり、4×2kバイトの容量を有する。各ウェイは、16個のキャッシュエントリを有する。

図3に1つのキャッシュエントリにおける詳細なビット構成を示す。同図のように、1つのキャッシュエントリは、バリッドフラグV0~V3、21ビットのタグ、128バイトのラインデータ、ウィークフラグW、ダーティフラグD0~D3を有する。

## 【0025】

タグは21ビットのタグアドレスのコピーである。

ラインデータは、タグアドレスおよびセットインデックスにより特定されるブロック中の128バイトデータのコピーであり、32バイトの4つのサブラインからなる。

バリッドフラグV0~V3は、4つのサブラインに対応し、サブラインが有効か否かを示す。

## 【0026】

ウィークフラグWは、当該キャッシュエントリのアクセス順序が最も古いものとみなすためのフラグである。つまり、W=1のとき、プロセッサ1からは当該キャッシュエントリにこれ以上読み出しも書き込みもなされないこと、あるいはアクセス頻度が低いことを意味する。また、W=1のとき、キャッシュメモリ3においては、リプレース制御に



関してアクセス順序が最古であることと取り扱うこと、つまり、最弱（ウィーク）のキャッシュエントリーであることを意味する。W=0 のとき、そうでないことを意味する。

#### 【0027】

ダーティフラグD0～D3は、4つのサブラインに対応し、そのサブラインにプロセッサから書き込みがあったか否か、つまりサブライン中にキャッシュされたデータが存在するが書き込みによりメモリ中のデータと異なるためメモリに書き戻すことが必要か否かを示す。

比較器32aは、アドレスレジスタ20中のタグアドレスと、セットインデックスにより選択されたセットに含まれる4つのタグ中のウェイ0のタグとが一致するか否かを比較する。比較器32b～32cについても、ウェイ31b～31dに対応すること以外は同様である。

#### 【0028】

アンド回路33aは、バリッドフラグと比較器32aの比較結果とが一致するか否かを比較する。この比較結果をh0とする。比較結果h0が1である場合は、アドレスレジスタ20中のタグアドレスおよびセットインデックスに対応するラインデータが存在すること、つまりウェイ0においてヒットしたことを意味する。比較結果h0が0である場合は、ミスヒットしたことを意味する。アンド回路33b～33dについても、ウェイ31b～31dに対応すること以外は同様である。その比較結果h1～h3は、ウェイ1～3でヒットしたかミスしたかを意味する。

#### 【0029】

オア回路34は、比較結果h0～h3のオアをとる。このオアの結果をhitとする。hitは、キャッシュメモリにヒットしたか否かを示す。

セレクトア35は、選択されたセットにおけるウェイ0～3のラインデータのうち、ヒットしたウェイのラインデータを選択する。

セレクトア36は、セレクトア35により選択された32ワードのラインデータのうち、ワードインデックスに示される1ワードを選択する。

#### 【0030】

デマルチプレクサ37は、キャッシュエントリーにデータを書き込む際に、ウェイ0～3の1つに書き込みデータを出力する。この書き込みデータはワード単位でよい。

制御部38は、キャッシュメモリ3の全体の制御を行う。特に、Wフラグの設定とWフラグに従うリプレース制御を行う。

#### 【0031】

##### <制御部の構成>

図4は、制御部38の構成を示すブロック図である。同図のように、制御部38は、リプレース部39とWフラグ設定部40を含む。

#### 【0032】

リプレース部39は、キャッシュミスによるリプレースに際して、W=1が設定されているキャッシュエントリーが存在すれば、当該キャッシュエントリーのアクセス順序が最も古いとみなしてリプレース対象として選択し、リプレースを行う。

Wフラグ設定部40は、プロセッサ1からのコマンドに応じてウィークフラグWを設定する。プロセッサ1は、もはや書き込みも読み出しもしないキャッシュエントリーについてウィークフラグWの設定を指示するコマンドをキャッシュメモリ3に対して発行する。

#### 【0033】

##### <Wフラグ設定部の構成>

図5は、Wフラグ設定部40の構成例を示すブロック図である。同図のようにWフラグ設定部40は、コマンドレジスタ401、スタートアドレスレジスタ402、サイズレジスタ403、加算器404、スタートアライナ405、エンドアライナ406、フラグ書換部407を備える。

#### 【0034】

コマンドレジスタ401は、プロセッサ1から直接アクセス可能なレジスタであり、プ

ロセッサ1により書き込まれたWフラグ設定コマンドを保持する。図6(c)に、コマンドレジスタ401にコマンドを書き込む命令の一例を示す。この命令は、通常の転送命令(mov命令)であり、ソースオペランドとしてコマンドを、デスティネーションオペランドとしてコマンドレジスタ(CR)を指定している。図6(d)に、コマンドの一例を示す。このコマンドは、Wフラグ設定コマンドを示す特定のコードである。Wフラグ設定コマンドは、スタートアドレスレジスタ402に保持されたスタートアドレスからサイズレジスタ403に保持されたサイズのアドレス範囲に対応するデータを保持するキャッシュエントリに対して、Wフラグを設定することを指示するコマンドである。

#### 【0035】

スタートアドレスレジスタ402は、プロセッサ1から直接アクセス可能なレジスタであり、プロセッサ1により書き込まれたスタートアドレスを保持する。このスタートアドレスはWフラグを設定すべきアドレス範囲の開始位置を示す。図6(a)に、スタートアドレスレジスタ402にスタートアドレスを書き込む命令の一例を示す。この命令も、図6(c)と同様に通常の転送命令(mov命令)である。

#### 【0036】

サイズレジスタ403は、プロセッサ1から直接アクセス可能なレジスタであり、プロセッサ1により書き込まれたサイズを保持する。このサイズは、スタートアドレスからのアドレス範囲を示す。図6(b)に、サイズレジスタ403にサイズを書き込む命令の一例を示す。この命令も、図6(c)と同様に通常の転送命令(mov命令)である。なお、サイズの単位は、バイト数であっても、ライン数(キャッシュエントリ数)であってもよく、予め定められた単位であればよい。

#### 【0037】

加算器404は、スタートアドレスレジスタ402に保持されたスタートアドレスとサイズレジスタ403に保持されたサイズとを加算する。加算結果は、アドレス範囲の終了位置を指すエンドアドレスである。加算器404は、サイズがバイト数指定の場合はバイトアドレスとして加算し、サイズがライン数指定の場合はラインアドレスとして加算すればよい。

#### 【0038】

スタートアライナ405は、スタートアドレスをライン境界の位置に調整する。この調整によりプロセッサ1はラインサイズ及びライン境界とは無関係に任意のアドレスをスタートアドレスとして指定することができる。

エンドアライナ406は、エンドアドレスをライン境界の位置に調整する。この調整によりプロセッサ1はラインサイズ及びライン境界とは無関係に任意の大きさを上記サイズとして指定することができる。

#### 【0039】

図7に、スタートアライナ405及びエンドアライナ406の説明図を示す。同図において、プロセッサ1から指定されたスタートアドレスはラインNの途中の任意の位置を指す。スタートアライナ405は、次のライン(N+1)の先頭を指すよう調整し、調整後のアドレスをアラインスタートアドレスとして出力する。アラインスタートアドレスが指すラインをスタートラインと呼ぶ。

#### 【0040】

また、エンドアドレスはラインMの途中の任意の位置を指す。エンドアライナ406は、直前のライン(M-1)の先頭を指すよう調整し、調整後のアドレスをアラインエンドアドレスとして出力する。アラインエンドアドレスが指すラインをエンドラインと呼ぶ。

この場合、スタートライン(ライン(N+1))からエンドライン(ライン(M-1))までの各ライン(キャッシュエントリ)にWフラグが設定されることになる。このように、スタートアライナ405及びエンドアライナ406がプロセッサ1から指定されたスタートアドレスからエンドアドレスまでのアドレス範囲よりも内側にアラインしているのは、ラインNとラインMの外側の部分にはプロセッサ1から書き込み又は読み出しが発生する可能性があるからである。

## 【0041】

フラグ書換部 407 は、アラインスタートアドレスが指すラインからアラインエンドアドレスが指すラインまで（図 7 の例ではライン (N+1) からライン (M-1) まで）、キャッシュメモリ 3 にエンタリーされていれば W フラグを 1 に設定する。

## 【0042】

## &lt;W フラグ設定処理&gt;

図 8 は、フラグ書換部 407 における W フラグ設定処理の一例を示すフローチャートである。

フラグ書換部 407 は、コマンドレジスタ 401 に W フラグ設定コマンドが保持されている場合、スタートラインからエンドラインまでの各ラインアドレスを順に出力しながらループ 1 の処理 (S82~S86) を行う。フラグ書換部 407 は、各ラインについて同じ処理を行うので、ここでは 1 ライン分の処理について説明する。

すなわち、フラグ書換部 407 は、キャッシュメモリ 3 がプロセッサ 1 からアクセスされていない間に、ラインアドレスをアドレスレジスタ 20 に出力し (S83)、アドレスレジスタ 20 のタグアドレスとキャッシュエントリーのタグとを比較器 32a~32d に比較させ、ヒットするかどうかを判定する (S84)。さらにフラグ書換部 407 は、ヒットした場合には、ヒットしたキャッシュエントリーに対して W フラグを 1 にセットし (S85)、ミスヒットした場合には、キャッシュメモリにエンタリーされていないのでなにもしない。

これにより、スタートラインからエンドラインまでの各ラインについて、キャッシュメモリ 3 にエンタリーされている場合には、W フラグが 1 に設定される。

## 【0043】

## &lt;リプレース処理&gt;

図 9 は、リプレース部 39 におけるリプレース処理を示すフローチャートである。同図においてリプレース部 39 は、メモリアクセスがミスしたとき (ステップ S91)、セットインデックスにより選択されたセットにおける、4 つウェイのウィークフラグ W を読み出し (ステップ S92)、4 つのウィークフラグの論理和が 1 であるか否かつまり W=1 のウェイが存在すれか否かを判定する (ステップ S93)。W=1 のウェイが存在すると判定された場合、当該キャッシュエントリーのアクセス順序が最も古いとみなして W=1 のウェイを 1 つ選択し (ステップ S94)、W=1 のウェイが存在しないと判定された場合、通常の LRU 方式によりウェイを 1 つ選択する (ステップ S95)。このとき、ウィークフラグ W が 1 になっているウェイが複数存在する場合は、リプレース部 39 はランダムに 1 つを選択する。

## 【0044】

さらに、リプレース部 39 は、当該セットにおける選択されたウェイのキャッシュエントリーを対象にリプレースし (ステップ S96)、リプレース後に当該キャッシュエントリーのウィークフラグ W を 0 初期化する (ステップ S97)。なお、このときバリッドフラグ V、ダーティフラグ D は、それぞれ 1、0 に初期化される。

このように、W=1 のウェイが存在しない場合、リプレース対象は、通常の LRU 方式により選択される。また、W=1 のウェイが存在する場合、リプレース対象は、W=1 のウェイのアクセス順序が最も古いものと取り扱われる結果、W=1 のウェイのキャッシュエントリーが選択される。これにより、アクセス頻度の低い W=1 のデータがキャッシュメモリに存在するために生じるキャッシュミスは低減することができる。

## 【0045】

以上説明してきたように、本実施の形態におけるキャッシュメモリによれば、ウィークフラグ W=1 のラインは、プロセッサからこれ以上読み出しも書き込みもなされないラインであり、そのアクセス順序が最古として取り扱われる結果、真っ先にリプレース対象として選択される。これによりアクセス頻度の低いデータによるキャッシュミスの誘発を低減することができる。

## 【0046】

また、従来のLRU方式におけるアクセス順序を示す順序データを直接改変することなく、間接的にWフラグを付加することによってアクセス順序を改変しているのが、複雑なハードウェア回路を追加することなく、実現することができる。

#### 【0047】

##### <変形例>

なお、本発明のキャッシュメモリは、上記の実施の形態の構成に限るものではなく、種々の変形が可能である。以下、変形例のいくつかについて説明する。

(1) 使用フラグUを用いる擬似LRUの代わりに、4つのウェイのアクセス順序を示す順序データをキャッシュエントリ毎に保持及び更新して、従来通りのLRU方式でリプレース対象を選択する構成としてもよい。この場合も、W=1のキャッシュエントリを

アクセス順序に関わらず、真っ先にリプレース対象として選択するようにすればよい。さらに、上記実施の形態では、Wフラグの付加により間接的に順序データを改変しているが、順序データを直接改変する構成としてもよい。

(2) 上記実施の形態では、ウィークフラグWによりアクセス順序が最古であることを示しているが、アクセス順序が最新又は最古でないことを示すものとしてもよい。この場合、リプレース部39は、W=1のキャッシュエントリは最古でないこととみなして、リプレース対象として選択せず、他のキャッシュエントリを選択する構成とすればよい。最古でないことを示すウィークフラグWを、アクセス頻度の高いデータあるいはアクセス頻度が中くらいのデータを保持するキャッシュエントリに付加することにより、無駄リプレースを防止することができる。

(3) プロセッサ1が、ウィークフラグW=1の設定とデータの書き込みとを命令する特別なストア命令を実行し、制御部38は、さらに、特別なストア命令を検出する命令検出部と、当該ストア命令による書き込みの際にW=1に設定するフラグ設定部とを備える構成としてもよい。

(4) 上記実施の形態では、4ウェイ・セット・アソシエイティブのキャッシュメモリを例に説明したが、ウェイ数は、いくつでもよい。また、上記実施の形態では、セット数が16である例を説明したが、セット数はいくつでもよい。

(5) 上記実施の形態では、セット・アソシエイティブのキャッシュメモリを例に説明したが、フル・アソシエイティブ方式のキャッシュメモリであってもよい。

(6) 上記実施の形態では、サブラインのサイズをラインのサイズの1/4としているが、1/2、1/8、1/16等他のサイズでもよい。その場合、各キャッシュエントリは、サブラインと同数のバリッドフラグおよびダーティフラグをそれぞれ保持すればよい。

#### 【0048】

##### (実施の形態2)

実施の形態1では、通常のLRU方式を前提としてウィークフラグWによってアクセス順序を最古化する構成について説明した。本実施の形態では、通常のLRU方式とは異なりアクセス順序を示す順序データを1ビットのフラグで表した擬似的なLRU方式と、ウィークフラグによる最古化とを行うキャッシュメモリについて説明する。

#### 【0049】

##### <キャッシュメモリの構成>

図10は、本発明の実施の形態2におけるキャッシュメモリの構成を示すブロック図である。同図のキャッシュメモリは、図2の構成と比較して、ウェイ31a~31dの代わりにウェイ131a~131dを備える点と、制御部38の代わりに制御部138を備える点とが異なっている。以下、同じ点は説明を省略して、異なる点を中心に説明する。

#### 【0050】

ウェイ131aは、ウェイ31aと比べて、各キャッシュエントリ中に、使用フラグUが追加されている点異なる。ウェイ131b~131dも同様である。使用フラグUは、4つのウェイ間でのアクセス順序を示す順序データの代わりに設けられ、アクセス順序

を1ビットで表現するフラグである。

図11に、キャッシュエントリーのビット構成を示す。同図のビット構成は、図3と比較して、使用フラグUが追加されている点が異なる。

#### 【0051】

使用フラグUは、そのキャッシュエントリーにアクセスがあったか否かを示し、LRU方式におけるミスヒットによるリプレースに際して4つのウェイのキャッシュエントリーにおけるアクセス順序データの代わりに用いられる。より正確には、使用フラグUの1は、アクセスがあったことを、0はないことを意味する。ただし、1つのセット内の4つウェイの使用フラグが全て1になれば、0にリセットされる。別言すれば、使用フラグUは、アクセスされた時期が古いか新しいか2つの相対的な状態を示す。つまり、使用フラグUが1のキャッシュエントリーは、使用フラグが0のキャッシュエントリーよりも新しくアクセスされたことを意味する。

制御部138は、制御部38と比べて、LRU方式におけるアクセス順序情報の代わりに使用フラグUを用いてリプレース制御する点が異なる。

#### 【0052】

##### <制御部の構成>

図12は、制御部138の構成を示すブロック図である。同図の制御部138は、制御部38と比較して、リプレース部39の代わりにリプレース部139を備える点と、フラグ更新部41が追加された点とが異なる。

#### 【0053】

リプレース部139は、使用フラグUをアクセス順序とする擬似的なLRU方式によりキャッシュミス時にリプレース処理を行う。その際、リプレース部139は、ウィークフラグWが1のキャッシュエントリーがあれば、最古のキャッシュエントリーとして扱い、真っ先にリプレース対象として選択する。

フラグ更新部41は、キャッシュメモリがアクセスされたとき使用フラグUの更新処理を行う。

#### 【0054】

##### <使用フラグUの説明>

図13は、リプレース部139による使用フラグの更新例を示す。同図の上段、中段、下段は、ウェイ0～3に跨るセットNを構成する4つのウェイのキャッシュエントリーを示している。4つのキャッシュエントリー右端の1又は0は、それぞれ使用フラグの値である。この4つの使用フラグUをU0～U3と記す。

#### 【0055】

同図上段では(U0～U3)=(1、0、1、0)であるので、ウェイ0、2のキャッシュエントリーはアクセスがあったことを、ウェイ1、3のキャッシュエントリーはアクセスがないことを意味する。

この状態で、メモリアクセスがセットN内のウェイ1のキャッシュエントリーにヒットした場合、同図中段に示すように、(U0～U3)=(1、1、1、0)に更新される。つまり、実線に示すようにウェイ1の使用フラグU1が0から1に更新される。

#### 【0056】

さらに、同図中段の状態、メモリアクセスがセットN内のウェイ3のキャッシュエントリーにヒットした場合、同図下段に示すように、(U0～U3)=(0、0、0、1)に更新される。つまり、実線に示すようにウェイ3の使用フラグU3が0から1に更新される。加えて、破線に示すようにウェイ3以外の使用フラグU0～U2が1から0に更新される。これにより、ウェイ3のキャッシュエントリーが、ウェイ0～2の各キャッシュエントリーよりも新しくアクセスされたことを意味することになる。

#### 【0057】

リプレース部139は、キャッシュミス時にW=1のキャッシュエントリーが存在しなければ、使用フラグに基づいてリプレース対象のキャッシュエントリーを決定してリプレースを行う。例えば、リプレース部139は、図5上段では、ウェイ1とウェイ3の何れ

かをリプレース対象と決定し、図5中段ではウェイ3をリプレース対象と決定し、図5下段ではウェイ0～2の何れかをリプレース対象と決定する。

#### 【0058】

##### <ウィークフラグWの説明>

図14(a)ウィークフラグが存在しないと仮定した場合の比較例であり、キャッシュエントリがリプレースされる様子を示す図である。同図においても、図13と同様にウェイ0～3に跨るセットNを構成する4つのキャッシュエントリを示している。、4つのキャッシュエントリ右端の1又は0は、それぞれ使用フラグの値である。また、データEのみアクセス頻度の低いデータを、データA、B、C、Dはアクセス頻度の高いデータとする。

#### 【0059】

同図(a)の第1段目の状態で、プロセッサ1がデータEにアクセスすると、キャッシュミスが発生する。このキャッシュミスにより、例えば、U=0のキャッシュエントリの中からアクセス頻度の高いデータCのキャッシュエントリがアクセス頻度の低いデータEにリプレースされ、第2段目の状態となる。

第2段目の状態で、プロセッサ1がデータCにアクセスすると、キャッシュミスが発生する。このキャッシュミスにより、U=0のキャッシュエントリであるアクセス頻度の高いデータDのキャッシュエントリがアクセス頻度の高いデータCにリプレースされ、第3段目の状態となる。

#### 【0060】

第3段目の状態で、プロセッサ1がデータDにアクセスすると、キャッシュミスが発生する。このキャッシュミスにより、例えば、アクセス頻度の高いデータCのキャッシュエントリがアクセス頻度の高いデータDにリプレースされ、第3段目の状態となる。

同様に、第4段目でも、使用頻度の低いデータEはリプレース対象として選択されないで、キャッシュメモリに残っている。

#### 【0061】

第5段目の状態で、使用頻度の低いデータEは最も古い(U=0)であることから、リプレース対象として選択されて、追い出される。

このように、擬似LRU方式において(通常のLRU方式においても)、アクセス頻度の低いデータEによって、4ウェイの場合は最悪4回のキャッシュミスを誘発する場合がある。

#### 【0062】

図14(b)は、リプレース処理におけるウィークフラグWの役割を示す説明図である。

同図(b)の第1段目の状態(同図(a)の第1段目と同じ)で、プロセッサ1がデータEにアクセスすると、キャッシュミスが発生する。このキャッシュミスにより、例えば、U=0のキャッシュエントリの中からアクセス頻度の高いデータCのキャッシュエントリがアクセス頻度の低いデータEにリプレースされる。このとき、プロセッサ1は、データEのキャッシュエントリにウィークフラグWを1に設定するものとする。これにより、次のキャッシュミス時にデータEのキャッシュエントリが真っ先に追い出され、第2段目の状態となる。

#### 【0063】

第2段目の状態で、プロセッサ1がデータCにアクセスすると、キャッシュミスが発生する。このキャッシュミスにより、W=1のキャッシュエントリであるアクセス頻度の低いデータEのキャッシュエントリがリプレース対象として選択され、アクセス頻度の高いデータCにリプレースされ、第3段目の状態となる。

このように、ウィークフラグWを設けることにより、アクセス頻度の低いデータによるキャッシュミスの誘発を低減することができる。

#### 【0064】

##### <Uフラグ更新処理>

図15は、フラグ更新部41におけるUフラグ更新処理を示すフローチャートである。同図では、バリッドフラグが0（無効）であるキャッシュエントリーの使用フラグUは0に初期化されているものとする。

#### 【0065】

同図において、フラグ更新部41は、キャッシュヒットしたとき（ステップS61）、セットインデックスにより選択されたセットにおけるヒットしたウェイの使用フラグUを1にセットし（ステップS62）、そのセット内の他のウェイの使用フラグUを読み出し（ステップS63）、読み出した使用フラグUが全て1であるか否かを判定し（ステップS64）、全て1でなければ終了し、全て1であれば他のウェイの全ての使用フラグUを0にリセットする（ステップS65）。

#### 【0066】

このようにしてフラグ更新部41は、図13、図14（a）（b）に示した更新例のように、使用フラグUを更新する。

#### 【0067】

##### <リプレース処理>

図16は、リプレース部139におけるリプレース処理を示すフローチャートである。同図においてリプレース部139は、メモリアクセスがミスしたとき（ステップS91）、セットインデックスにより選択されたセットにおける、4つウェイの使用フラグU及びウィークフラグWを読み出し（ステップS92）、W=1のウェイが存在するか否かを判定する（ステップS93）。W=1のウェイが存在しないと判定された場合、U=0のウェイを1つ選択する（ステップS94）。このとき、使用フラグUが0になっているウェイが複数存在する場合は、リプレース部139はランダムに1つを選択する。また、W=1のウェイが存在すると判定された場合、Uフラグの値に関わらずW=1のウェイを1つ選択する（ステップS95）。このとき、ウィークフラグWが1になっているウェイが複数存在する場合は、リプレース部139はランダムに1つを選択する。

#### 【0068】

さらに、リプレース部139は、当該セットにおける選択されたウェイのキャッシュエントリーを対象にリプレースし（ステップS96）、リプレース後に当該キャッシュエントリーの使用フラグUを1に、ウィークフラグWを0初期化する（ステップS97）。なお、このときバリッドフラグV、ダーティフラグDは、それぞれ1、0に初期化される。

このように、W=1のウェイが存在しない場合、リプレース対象は、使用フラグUが0のキャッシュエントリーの中から1つ選択される。

#### 【0069】

また、W=1のウェイが存在する場合、リプレース対象は、使用フラグUが0であると1であるとを問わず、W=1のウェイのキャッシュエントリーから1つ選択される。これにより図14（a）（b）に示したように、アクセス頻度の低いデータがキャッシュメモリに残ることによるキャッシュミスの誘発を低減することができる。

以上説明してきたように、本実施の形態におけるキャッシュメモリによれば、従来のLRU方式におけるアクセス順序を示すデータの代わりに1ビットの使用フラグを用いる擬似LRU方式を採用することにより、アクセス順序データとして1ビットのフラグでよいので、アクセス順序データのデータ量が少ないこと及び更新が簡単であることからハードウェア規模を小さくすることができる。

#### 【0070】

また、これ以上使用されないキャッシュエントリーにW=1が設定され、W=1のキャッシュエントリーが真っ先にリプレース対象として選択されるので、アクセス頻度の低いデータがキャッシュメモリに残ることによるキャッシュミスの誘発を低減することができる。

#### 【0071】

##### <変形例>

（1） 上記各実施形態において、プロセッサ1が、ウィークフラグWを1に設定しなが



らデータをアクセスするロード／ストア命令（以下W-L/S命令と略す）を実行し、制御部38又は制御部138は、W-L/S命令の実行を検出したときに、当該W-L/S命令によるアクセスの直後にWフラグを1に設定する構成としてもよい。図17に、その場合に制御部38又は138に備えられるWフラグ設定部40aの構成例を示す。

#### 【0072】

同図においてWフラグ設定部40aは、LD/ST命令検出部410、ウィーク指示検出部411、アンド回路412、フラグ書き換え部413とを備える。

LD/ST命令検出部410は、プロセッサ1がロード／ストア命令を実行されたことを検出する。ウィーク指示検出部411は、ロード／ストア命令の実行時にプロセッサ1からウィーク指示が出力されているかどうかを検出する。ウィーク指示は、例えば、プロセッサ1がからの信号線により検出することができる。アンド回路412は、ロード／ストア命令の実行が検出され、かつ、ウィーク指示が検出されたとき、フラグ書き換え部413に、W-L/S命令の検出を通知する。フラグ書き換え部413は、W-L/S命令が検出されたとき、当該W-L/S命令によりアクセスされたデータを保持するキャッシュエントリーに対してウィークフラグWを1に設定する。

#### 【0073】

(2) また、上記(1)においてWフラグ設定部40aは、実施の形態1における制御部38に設けられる場合には、ウィークフラグWの代わりに、アクセス順序を示す順序データを直接に改変する構成としてもよい。その場合、ウィーク指示検出部411は、設定すべきアクセス順序の指定付きロード／ストア命令を実行するプロセッサ1から、設定すべきアクセス順序を示す番号(N番目)を検出する。N番目は、4ウェイセットアソシイティブの場合1番から4番(又は0から3)の何れかでよい。例えば、プロセッサ1は、アクセス頻度の低いデータに対しては最古のN=4とし、アクセス頻度の低いデータに対してはN=1又は2番などを指定することができる。フラグ書き換え部413は、アクセス順序の指定付きロード／ストア命令によりアクセスされたデータを保持するキャッシュエントリーの順序データをN番に改変する。このように、順序データを直接に任意のN番に改変する構成としてもよい。

#### 【0074】

(3) 図5に示したWフラグ設定部40の代わりに図18に示すWフラグ設定部40bを備える構成としてもよい。Wフラグ設定部40bは、Wフラグ設定部40に対して比較器408を追加し、フラグ書換部407の代わりにフラグ書換部407aを備える構成となっている。比較器408は、加算器404から出力されるエンドラインのラインアドレスと、ウィークフラグ設定のためにフラグ書換部407aから出力され、タグアドレスレジスタ20に保持されたラインアドレスとが一致するか否かを判定する。この比較器408は、図8に示したWフラグ設定処理中のループ1におけるエンドアドレスの判定に用いられる。すなわち、フラグ書換部407aは、比較器408が一致すると判定した場合、ウィークフラグの設定を終了する。

#### 【0075】

(4) 図6(a)(b)(c)に示した各命令は、コンパイラによりプログラム中に挿入してもよい。その際、コンパイラは、例えば配列データの書き込みや、圧縮動画データをデコードする際のブロックデータの書き込み等、これ以上書き込みをしないプログラム位置に、上記各命令を挿入するようにすればよい。

#### 【産業上の利用可能性】

#### 【0076】

本発明は、メモリアクセスを高速化するためのキャッシュメモリに適しており、例えば、オンチップキャッシュメモリ、オフチップキャッシュメモリ、データキャッシュメモリ、命令キャッシュメモリ等に適している。

#### 【図面の簡単な説明】

#### 【0077】

【図1】 本発明の実施の形態1におけるプロセッサ、キャッシュメモリ、メモリを含



むシステムの概略構成を示すブロック図である。

【図 2】 キャッシュメモリの構成例を示すブロック図である。

【図 3】 キャッシュエントリーの詳細なビット構成を示す。

【図 4】 制御部の構成を示すブロック図である。

【図 5】 Wフラグ設定部の構成例を示すブロック図である。

【図 6】 (a) スタートアドレスレジスタにスタートアドレスを書き込む命令の一例を示す。(b) サイズレジスタにサイズを書き込む命令の一例を示す。(c) コマンドレジスタにコマンドを書き込む命令の一例を示す。(d) コマンドの一例を示す。

【図 7】 スタートアライナ及びエンドアライナの説明図を示す。

【図 8】 フラグ書換部における Wフラグ設定処理の示すフローチャートである。

【図 9】 リプレース部におけるリプレース処理を示すフローチャートである。

【図 10】 本発明の実施の形態 2 におけるキャッシュメモリの構成を示すブロック図である。

【図 11】 キャッシュエントリーのビット構成を示す。

【図 12】 制御部の構成を示すブロック図である。

【図 13】 リプレース部による使用フラグの更新例を示す。

【図 14】 (a) ウィークフラグが存在しない場合にキャッシュエントリーがリプレースされる様子を示す図である。(b) リプレース処理におけるウィークフラグ W の役割を示す説明図である。

【図 15】 フラグ更新部における Uフラグ更新処理を示すフローチャートである。

【図 16】 リプレース部におけるリプレース処理を示すフローチャートである。

【図 17】 Wフラグ設定部の他の構成例を示す図である。

【図 18】 Wフラグ設定部のさらに他の構成例を示す図である。

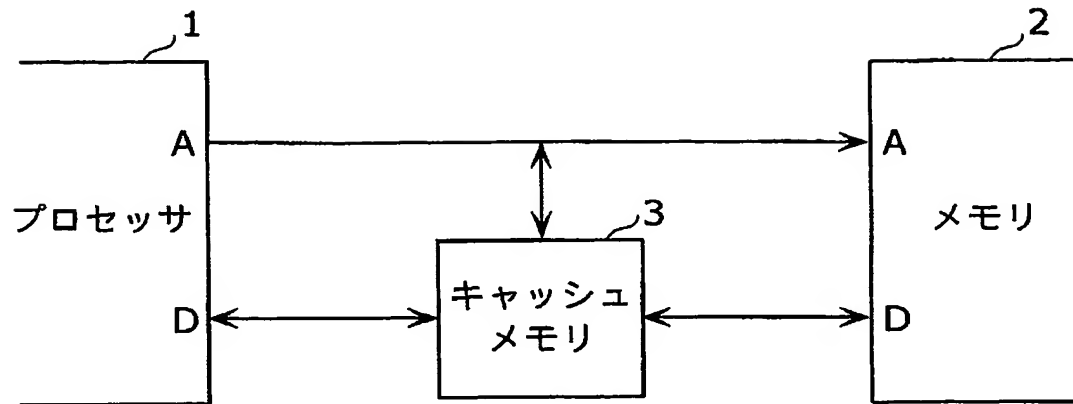
【符号の説明】

【0078】

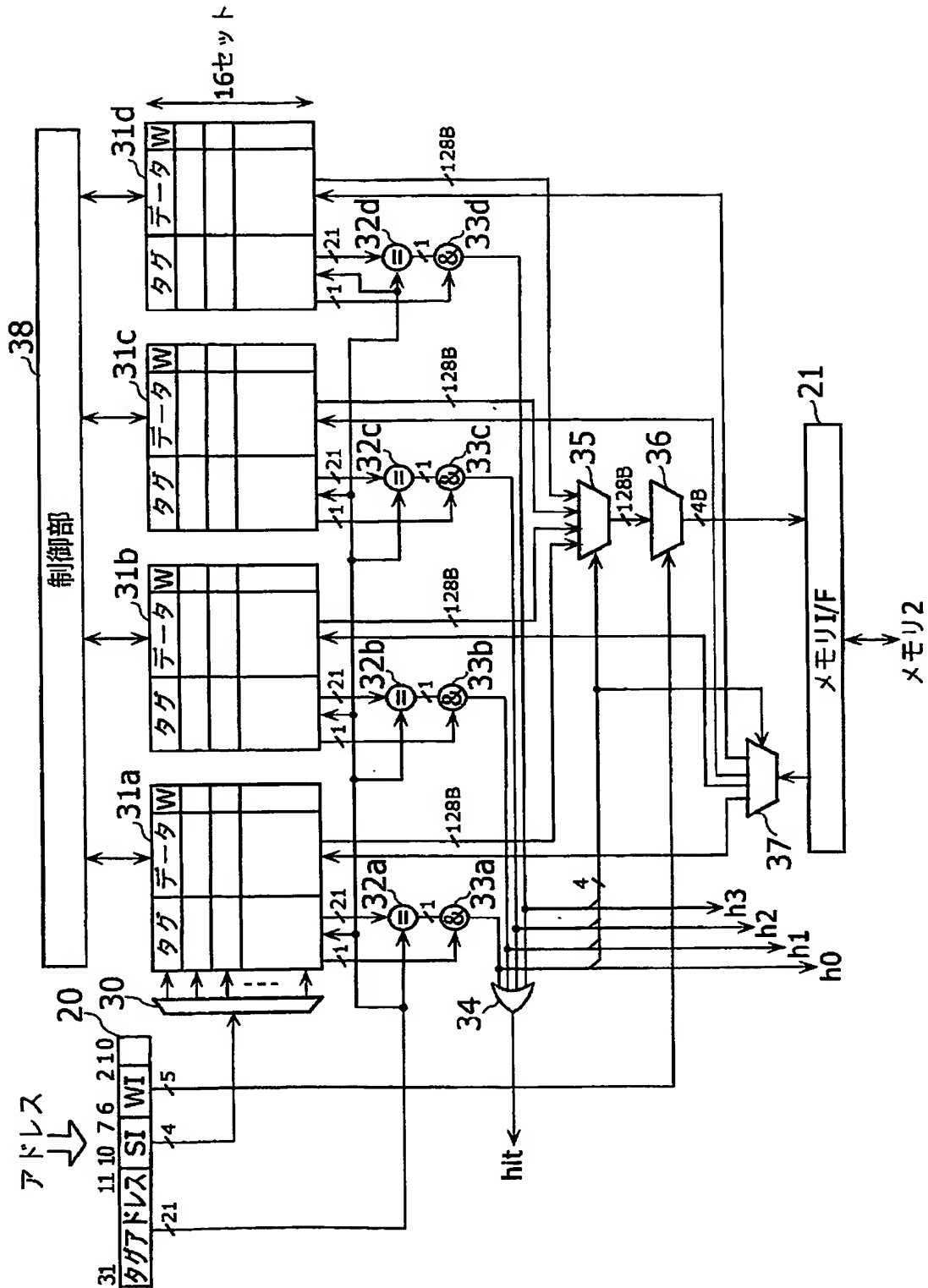
1	プロセッサ
2	メモリ
3	キャッシュメモリ
20	アドレスレジスタ
20	タグアドレスレジスタ
21	メモリ I/F
30	デコーダ
31a~31d	ウェイ
32a~32d	比較器
33a~33d	アンド回路
34	オア回路
35	セクタ
36	セクタ
37	デマルチプレクサ
38	制御部
39	リプレース部
40	Wフラグ設定部
41	フラグ更新部
131a~131d	ウェイ
138	制御部
139	リプレース部
401	コマンドレジスタ
402	スタートアドレスレジスタ
403	サイズレジスタ
404	加算器

4 0 5	スタートアライナ
4 0 6	エンドアライナ
4 0 7	フラグ書換部
4 0 7 a	フラグ書換部
4 0 8	比較器
4 1 0	LD / ST 命令検出部
4 1 1	ウィーク指示検出部
4 1 2	アンド回路
4 1 3	書き換え部

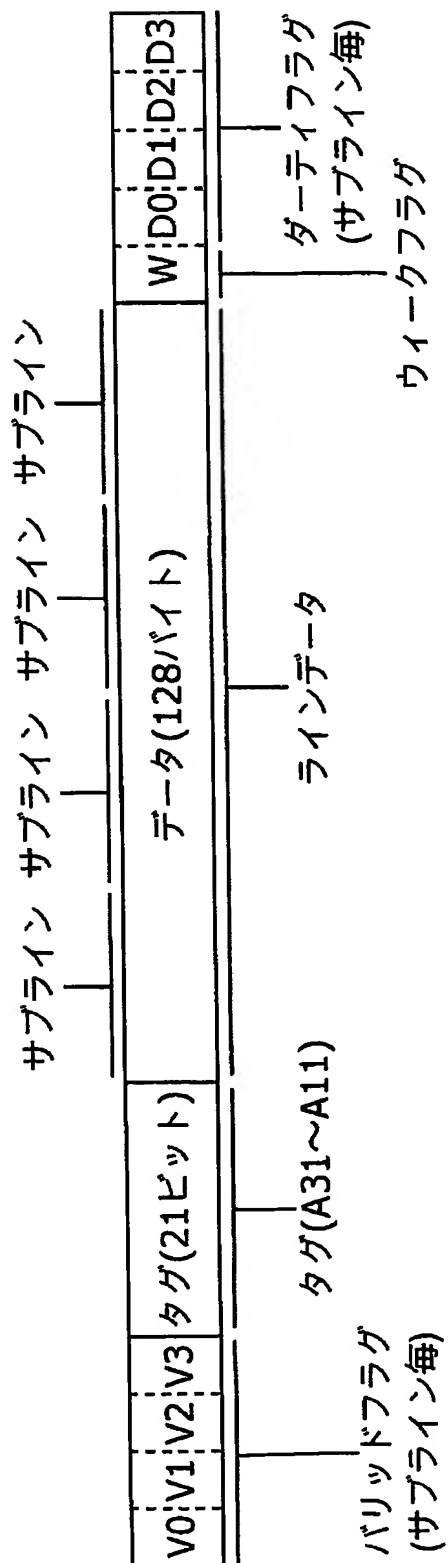
【書類名】図面  
【図1】



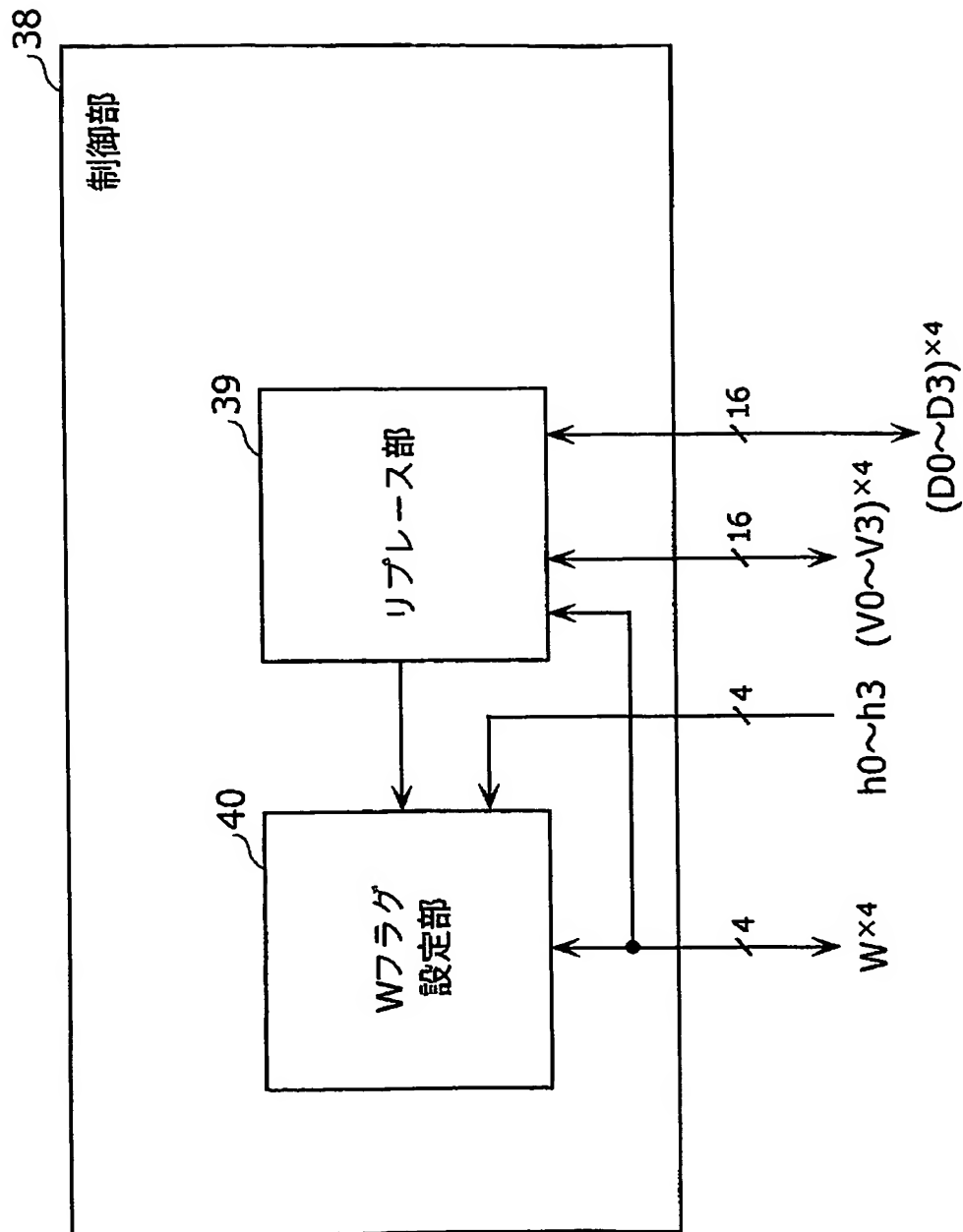
【図2】



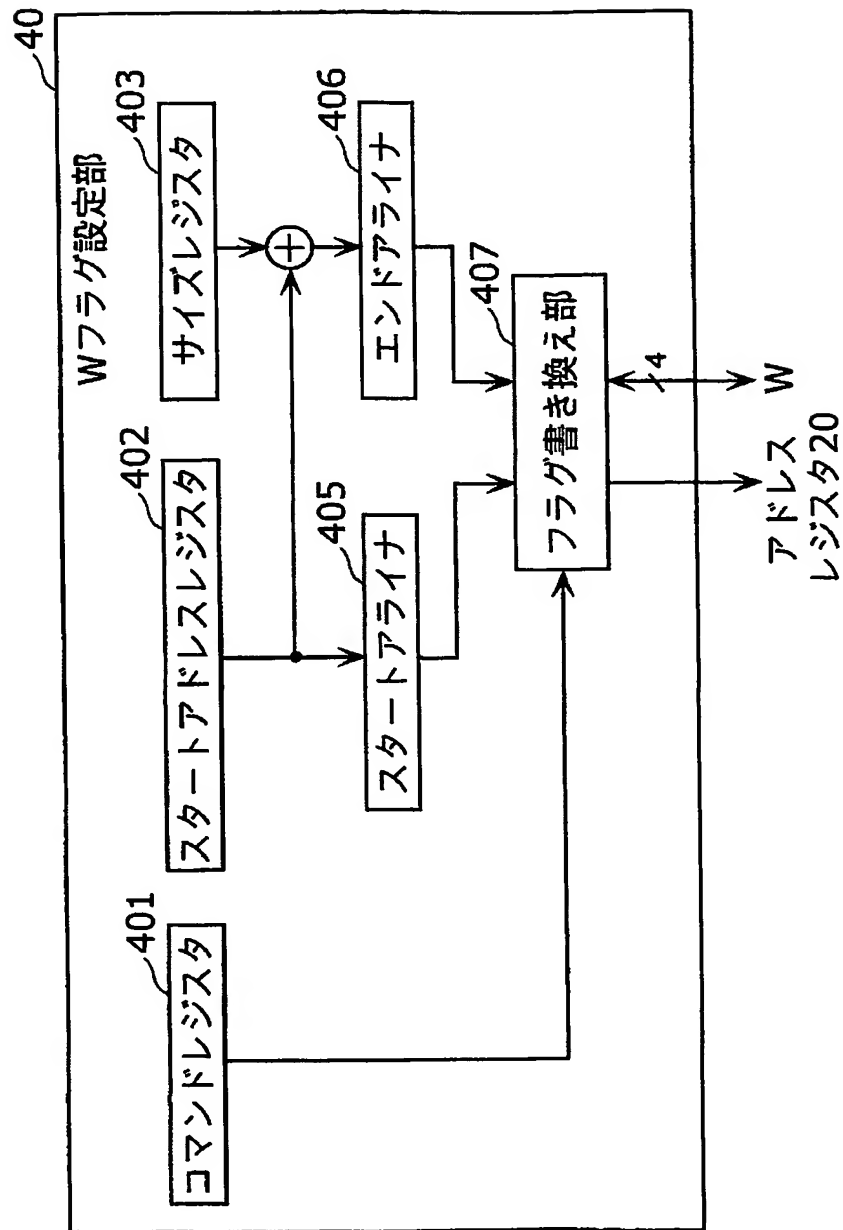
【図 3】



【図 4】



【図5】



【図 6】

(a)

```
mov SAR, start_adrs
```

└(ソース)スタートアドレスを指定

**-(デスティネーション)スタートアドレスレジスタを指定**

## 転送命令

(b)

mov    SR ,    size

## -(ソース)サイズを指定

-(デスティネーション)サイズレジスタを指定

## 轉送命令

(c)

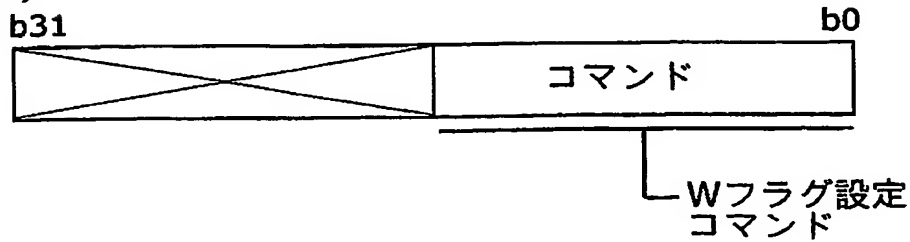
mov CR, command

-(ソース)コマンドを指定

### -(デスティネーション)コマンドレジスタを指定

## ・転送命令

(d)

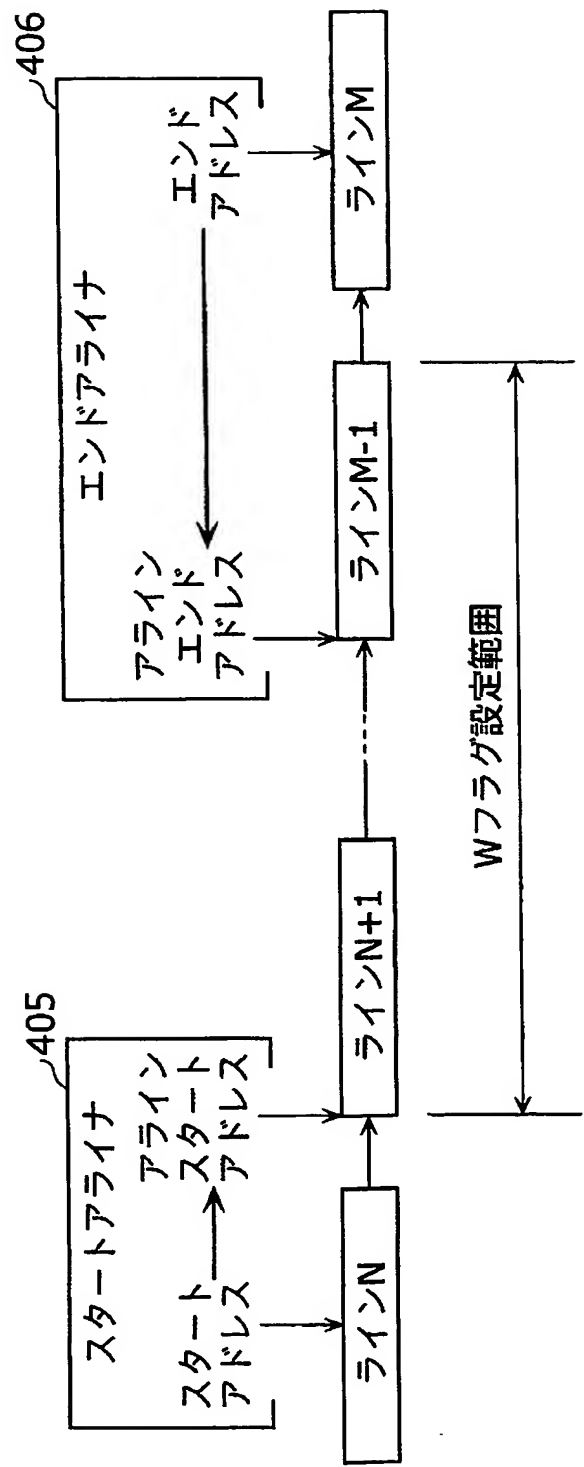


## コマンド

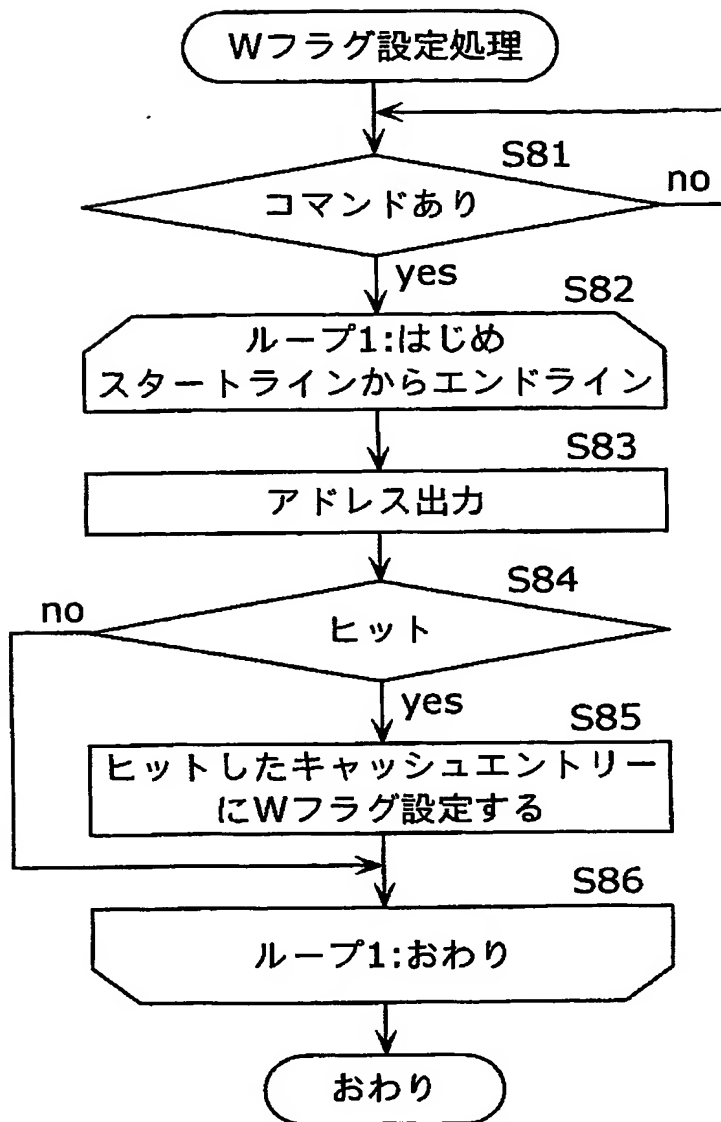
## Wフラグ設定 コマンド



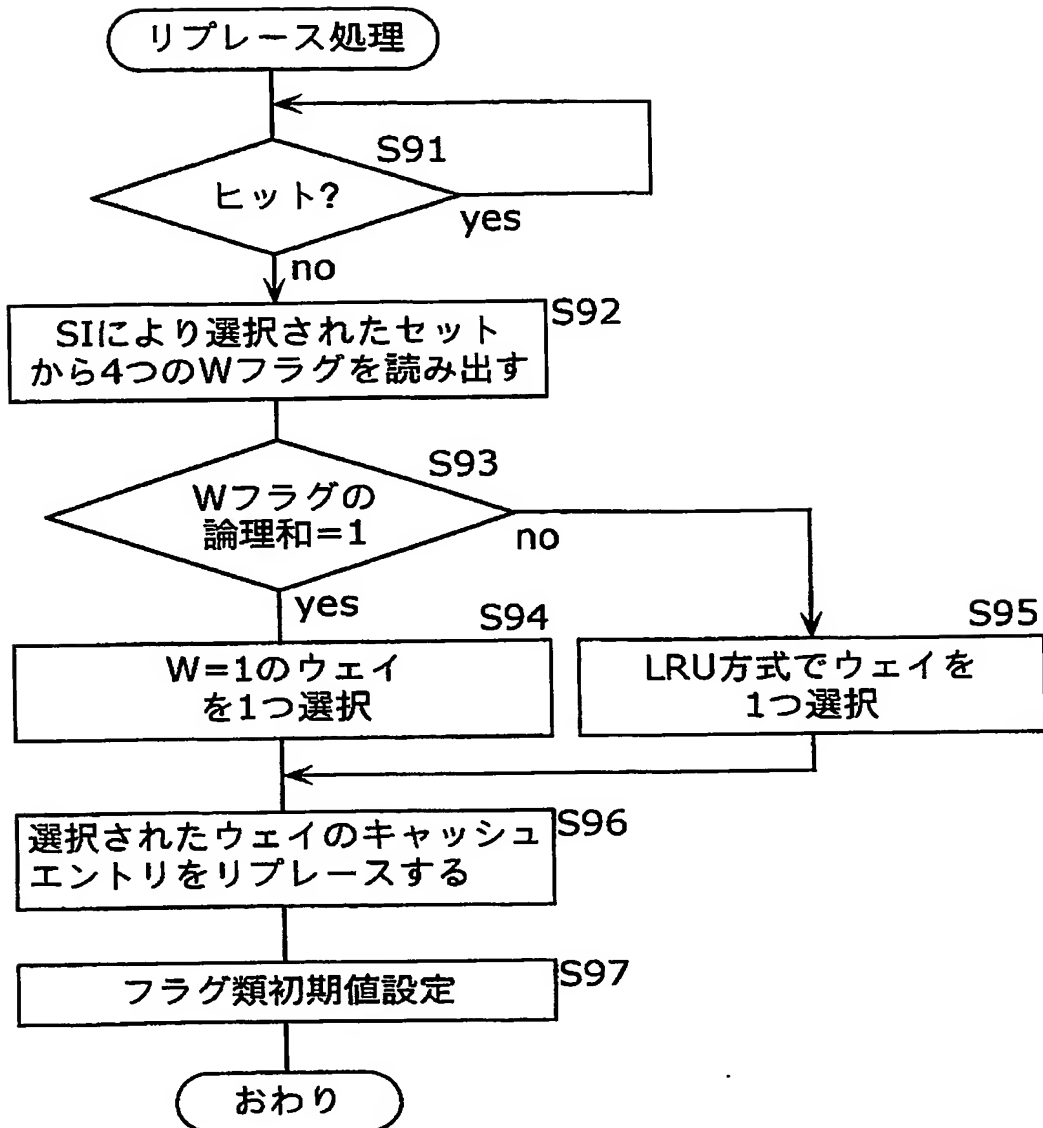
【図 7】



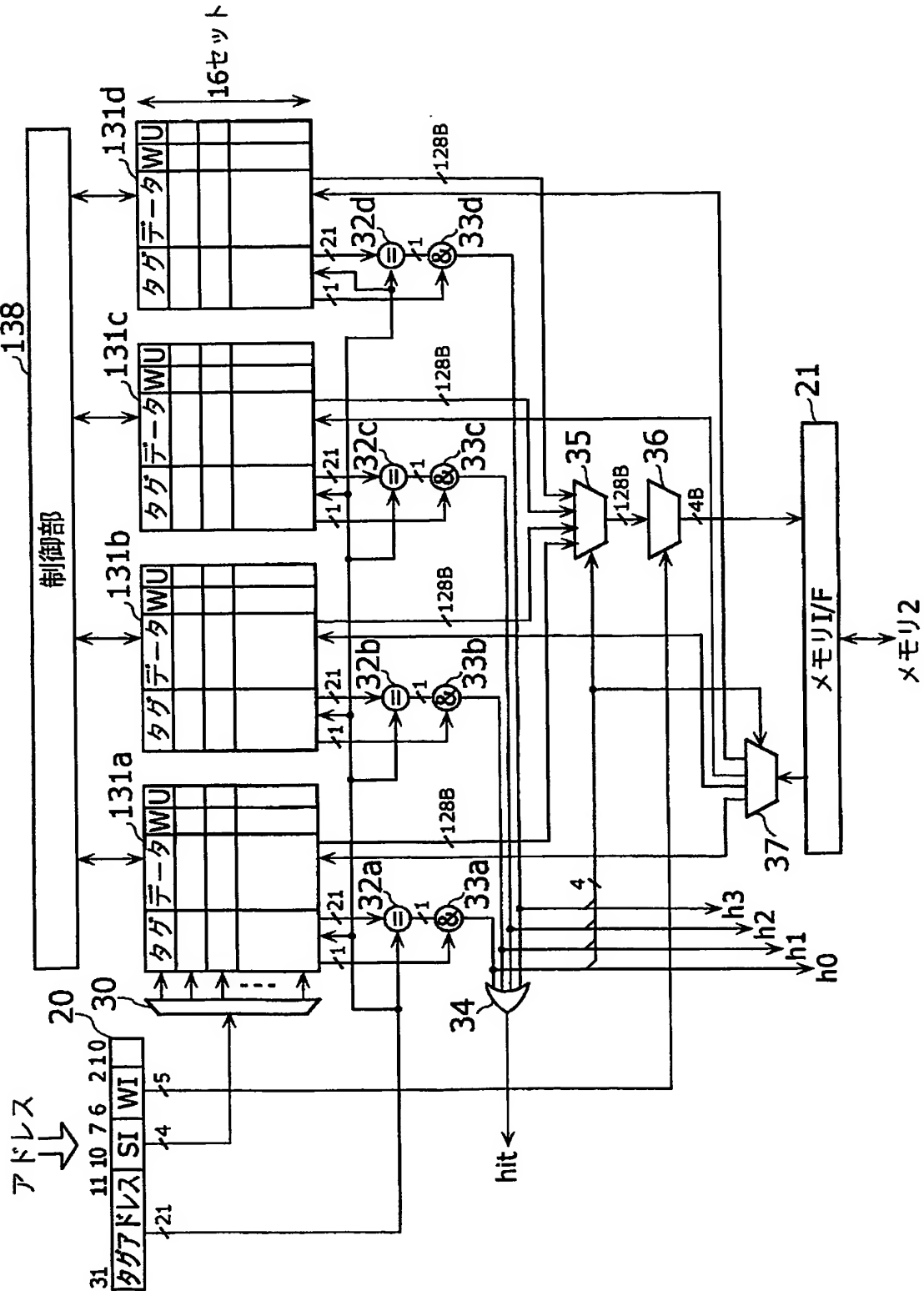
【図 8】



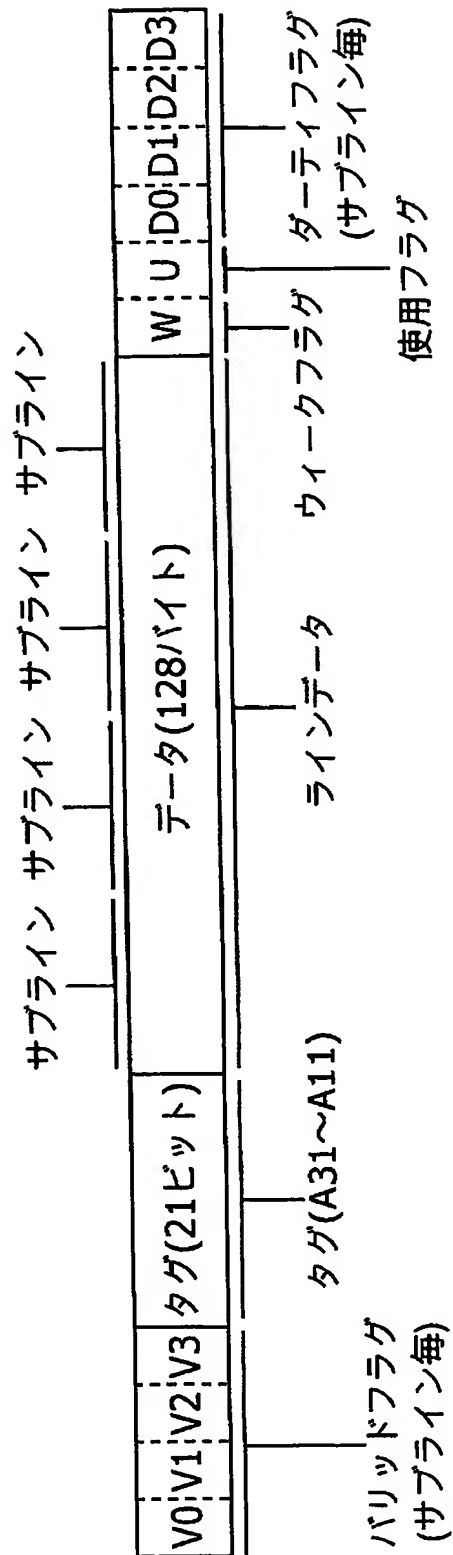
【図 9】



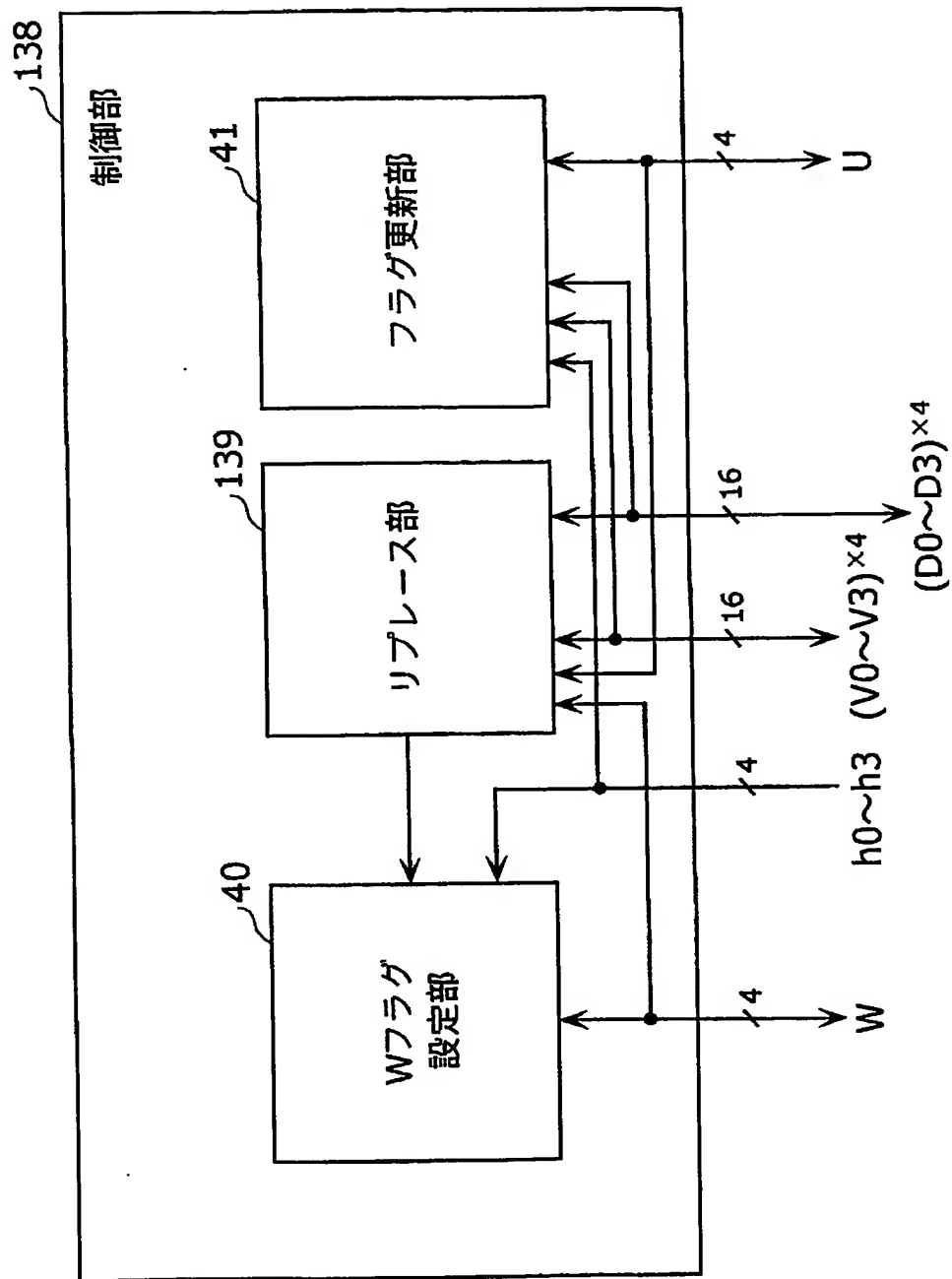
【図 10】



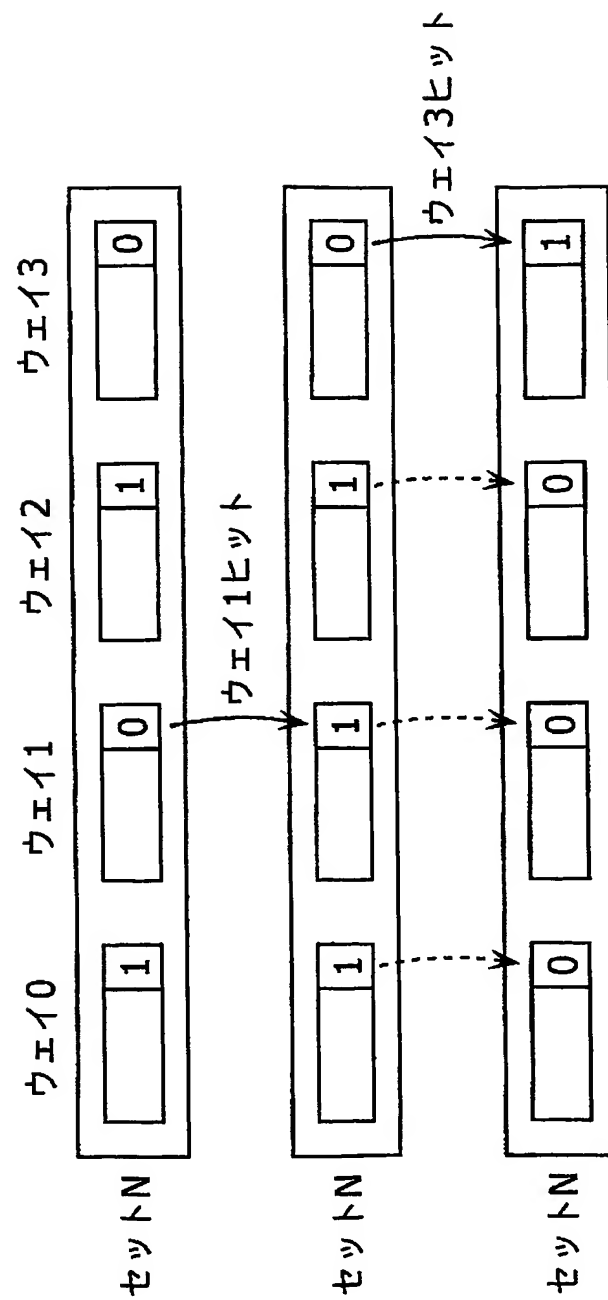
【図 11】



【図 12】

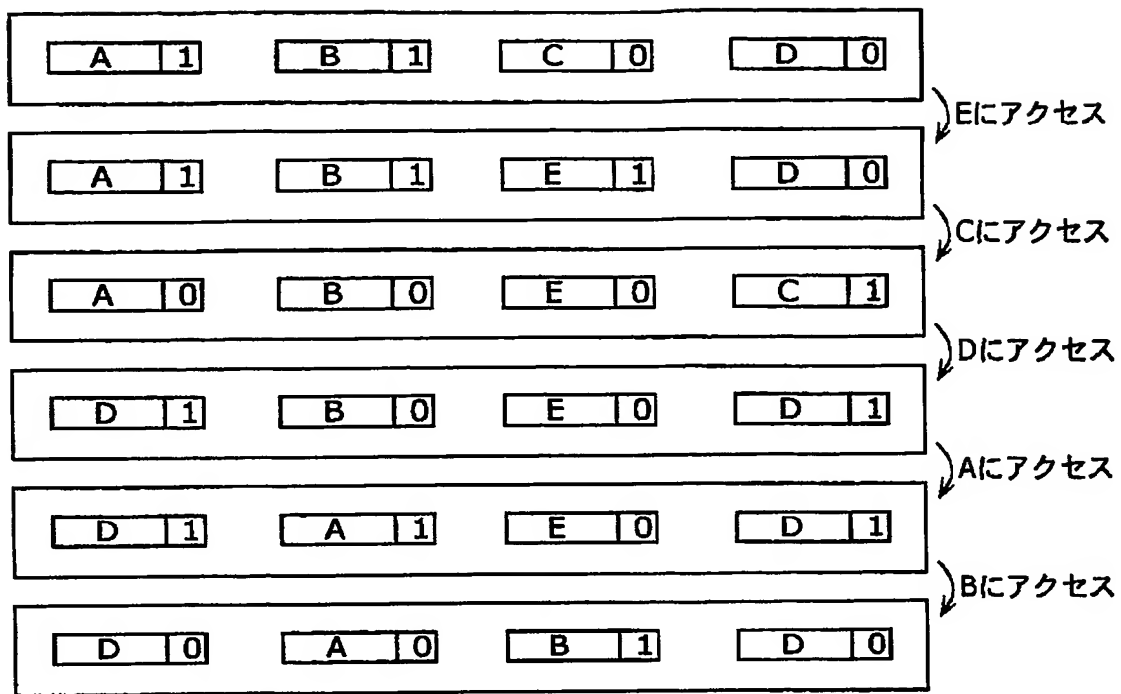


【図 13】

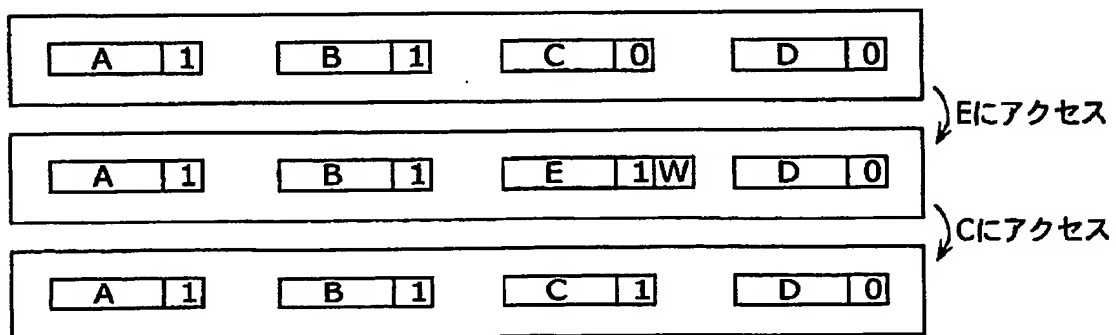


【図 14】

(a)

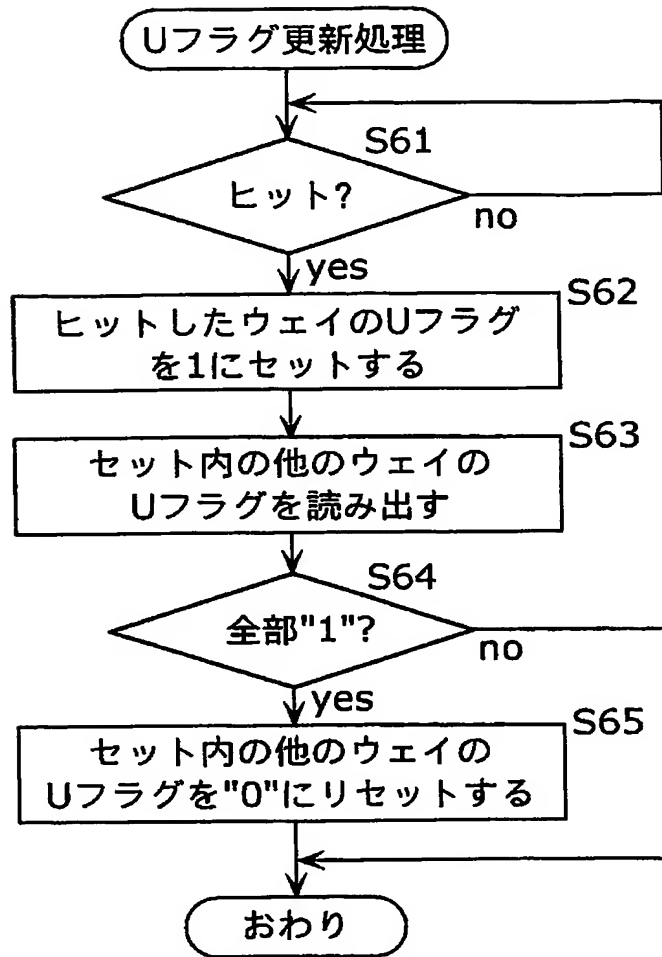


(b)

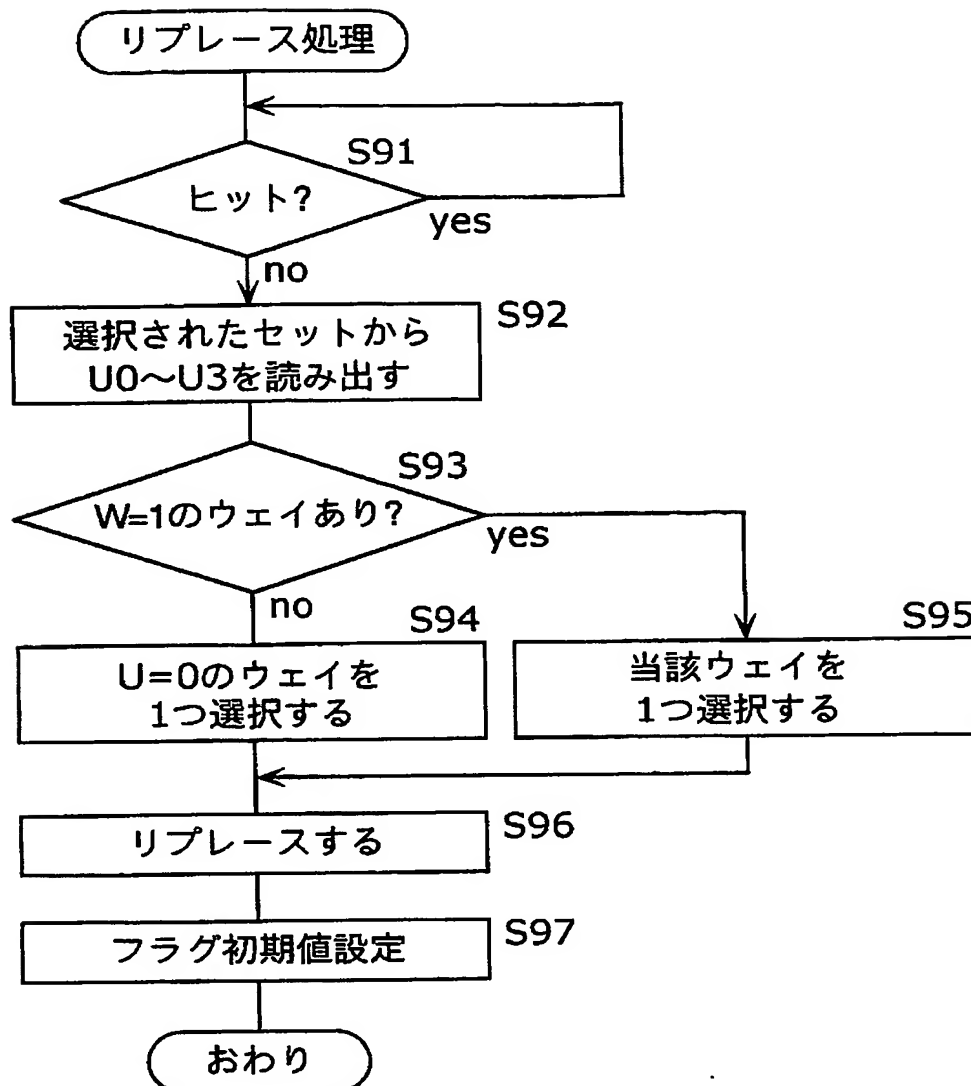




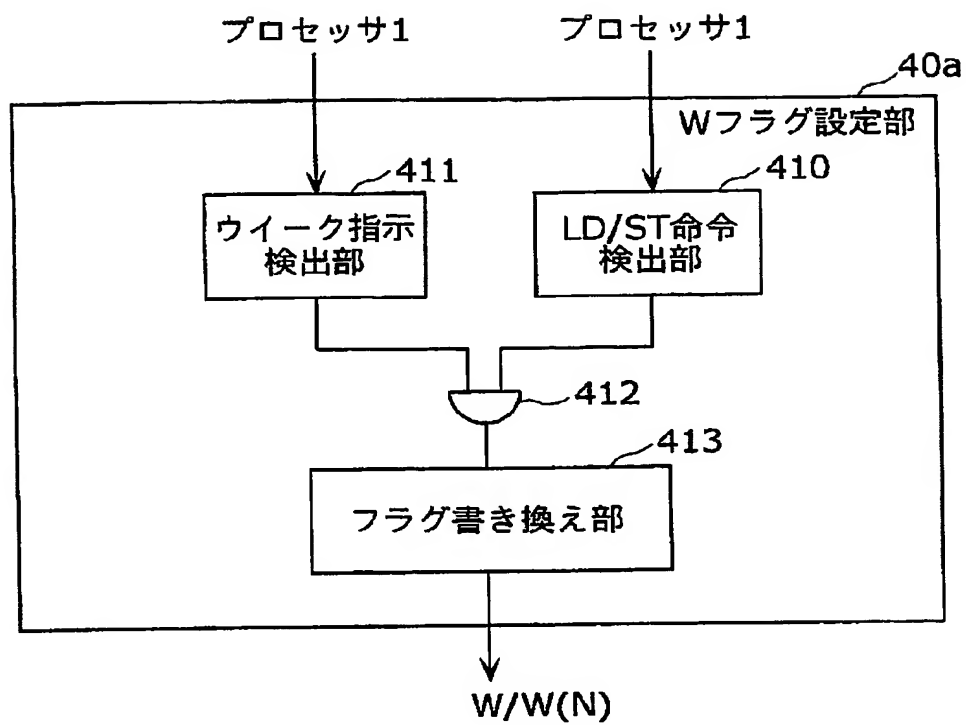
【図 15】



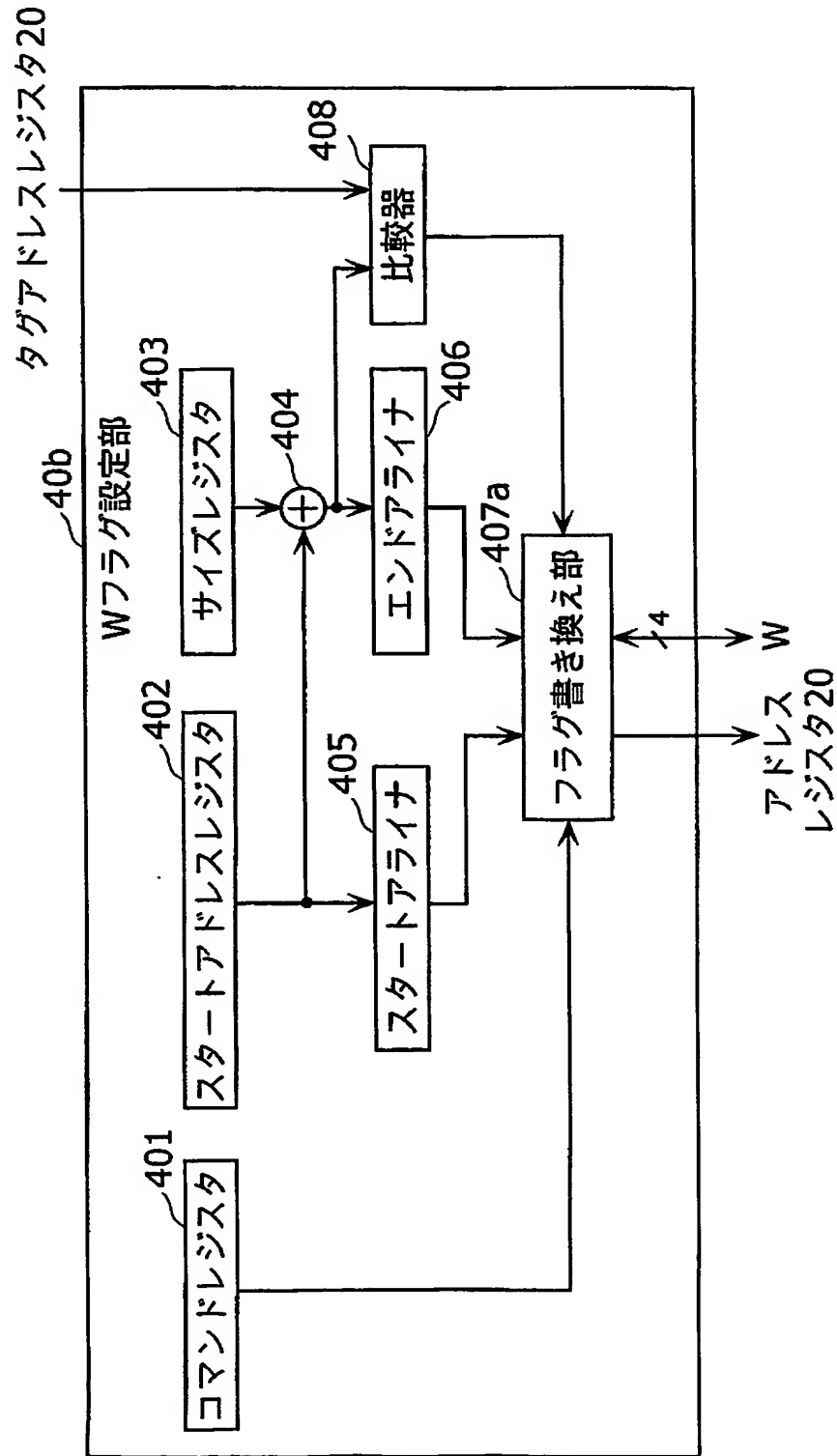
【図 16】



【図 17】



【図18】



【書類名】 要約書

【要約】

【課題】 アクセス頻度の低いデータを他のアクセス頻度の高いデータよりも優先的にリプレース対象とするキャッシュメモリを提供する。

【解決手段】 制御部 3 8 は、キャッシュの単位データを保持するキャッシュエントリー毎にアクセス順序を示す順序データを、実際のアクセス順序に反して改変する W フラグ設定部 4 0 と、改変後の順序データに基づいてリプレースすべきキャッシュエントリーを選択してリプレースするリプレース部 3 9 とを備える。

【選択図】 図 4

## 認定・付加情報

特許出願の番号	特願 2003-387351
受付番号	50301899682
書類名	特許願
担当官	第七担当上席 0096
作成日	平成15年11月19日

## &lt;認定情報・付加情報&gt;

【提出日】	平成15年11月18日
-------	-------------

特願 2 0 0 3 - 3 8 7 3 5 1

出 願 人 履 歴 情 報

識別番号 [ 0 0 0 0 0 5 8 2 1 ]

1. 変更年月日	1 9 9 0 年 8 月 2 8 日
[変更理由]	新規登録
住 所	大阪府門真市大字門真 1 0 0 6 番地
氏 名	松下電器産業株式会社

**This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record.**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

## **IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**